# APLIKASI JARINGAN SYARAF TIRUAN DALAM SISTEM DOCKING

TUGAS AKHIR

oleh :

A. AHMAD M. SIR.

NIM. 13288210/D



JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI BANDUNG

1 9 9 3

# APLIKASI JARINGAN SYARAF TIRUAN DALAM SISTEM DOCKING

TUGAS AKHIR

oleh :

A. AHMAD M. SIR.
NIM. 13288210/D



JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI BANDUNG
1993

# APLIKASI JARINGAN SYARAF TIRUAN DALAM SISTEM DOCKING

oleh :

**A. AHMAD M. SIR.**

NIM. 13288210/D

**TUGAS AKHIR**

Guna melengkapi syarat-syarat
untuk mencapai gelar Sarjana Teknik Elektro
di Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Bandung

Bandung, September 1993

# APLIKASI JARINGAN SYARAF TIRUAN
# DALAM SISTEM DOCKING

oleh :

**A. AHMAD M. SIR.**
NIM. 13288210/D

Tugas Akhir ini telah diterima dan disahkan
sebagai kolokium untuk memenuhi persyaratan
guna mencapai gelar

SARJANA TEKNIK ELEKTRO

pada

SUB JURUSAN TEKNIK ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI BANDUNG

disahkan oleh :

| Pembimbing I | Pembimbing II |
| --- | --- |
| Dr. Ir. Adang Suwandi Ahmad | Ir. S. Hakim Ad Dairi |
| NIP. 130 672 118 | NIP. 131 995 258 |

*Katakanlah "Kalau sekiranya lautan menjadi tinta untuk (menulis) kalimat-kalimat Tuhanku, sungguh habislah lautan itu sebelum habis (ditulis) kalimat–kalimat Tuhanku, meskipun Kami datangkan tambahan sebanyak itu".*

*(QS Al Kahfi : 109)*

***Kupersembahkan***
***untuk***
***Ayah Bunda tercinta***

# ABSTRAK

Pemrosesan sinyal sensor pada sistem docking sangat penting, karena ketepatan estimasi posisi dari hasil pengukuran tergantung pada hasil pemetaan dari masukan-keluarannya. Namun, proses pencarian hubungan matematis dari pemetaan tersebut amatlah rumit.

Pembentukan fungsi pemetaan tersebut di atas dapat dilakukan dengan teknik yang relatif lebih sederhana yaitu pendekatan pelatihan Jaringan Syaraf Tiruan (JST). Dengan kemampuannya untuk diajari, JST dapat dilatih dalam membentuk fungsi pemetaan pola masukan dari sejumlah sensor.

Pada tugas akhir ini diperlihatkan kemampuan JST sebagai sistem yang mampu belajar (*learning system*) dan hasil belajar yang diperoleh digunakan untuk memetakan pola masukan dari sejumlah sensor menjadi informasi yang diinginkan. Proses ini dipergunakan pada sistem docking.

# KATA PENGANTAR

*Alhamdulillah,* puji syukur kehadirat **Allah SWT** atas segala karunia yang telah diberikan-**Nya** sehingga penelitian tugas akhir ini dapat diselesaikan.

Juga tak lupa saya ucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah banyak membantu dalam penelitian dan penyusunan laporan tugas akhir ini baik secara moril maupun materil, terutama kepada :

1. Bapak **DR. Ir. Adang Suwandi Ahmad** dan Bapak **Ir. Syahirul Hakim Ad-Dairi**, yang telah membimbing penelitian tugas akhir ini.
2. Bapak **Prof. Kudrat Soemintapoera Ph.D**, sebagai kepala Laboratorium Sinyal dan Sistem yang telah memberikan izin pemakaian fasilitas dalam penelitian.
3. Bapak **Ir. Yudi Satria G**, yang banyak memberikan saran.
4. Seluruh Staf, dosen dan karyawan Laboratorium Sinyal dan Sistem atas segala bantuan moral dan materil.
5. Kedua **orang-tuaku**, serta kakak dan adik-adikku atas segala do,a dan dorongannya.
6. **Armet.**
7. **Yerry, Achdi, Daniel, Yusep, Nono, Juwi, Mbang, Sadmono, Sahat,**

**Nelson, Mas Eto, Mas Wicaksono dan Jonsen** atas segala kerjasamanya.

8. Semua pihak yang tak sempat disebutkan satu-persatu disini.

Penulis menyadari masih terdapat banyak kekurangan dalam penelitian ini. Oleh karena itu penulis sangat mengharapkan saran untuk menambahi kekurangan tersebut. Semoga laporan tugas akhir ini banyak memberikan manfaat kepada pembaca.

Bandung, September 1993

**A. Ahmad M.Siregar**

# DAFTAR ISI

# DAFTAR GAMBAR

# DAFTAR TABEL

# BAB I

# PENDAHULUAN

## 1.1 Latar Belakang Masalah

Sistem Docking adalah suatu sistem yang berfungsi untuk menempatkan suatu benda pada tempat yang diinginkan (sasaran). Untuk itu, sasaran (tempat benda) harus dijejaki terlebih dahulu, yang dalam kehidupan kita sehari-hari dilakukan dengan menggunakan mata manusia sebagai sensor. Pada sistem docking penjejakan sasaran dilakukan dengan menggunakan sensor. Hasil pengukuran sensor diproses menjadi informasi posisi benda oleh [illegible]. Informasi posisi benda kemudian diteruskan ke pengontrol [illegible] keputusan. Sinyal-sinyal komando yang diberikan pada aktuator untuk [illegible] sasaran dihasilkan oleh bagian pengontrol sistem docking. Jadi, ketepatan informasi posisi benda ikut menentukan ketepatan [illegible] untuk melakukan docking.

Mata manusia sebagai sensor, informasi yang diperoleh dari sasaran dipetakan menjadi sebuah informasi yang [illegible] secara matematis pemetaan tersebut sangat sulit untuk [illegible] yang diukur oleh mata tersebut sangat banyak. Masalah ini juga akan kita peroleh dalam mencari hubungan pemetaan pola masukan yang merupakan kombinasi besar hasil pengukuran dari sejumlah

# BAB I

# PENDAHULUAN

## 1.1 Latar Belakang Masalah

Sistem **Docking** adalah suatu sistem yang berfungsi untuk menempatkan suatu benda pada tempat yang diinginkan (sasaran). Untuk itu, sasaran (tempat benda) harus dijejaki terlebih dahulu, yang dalam kehidupan kita sehari-hari dilakukan dengan menggunakan mata manusia sebagai sensor. Pada sistem docking penjejakan sasaran dilakukan dengan menggunakan sensor. Hasil pengukuran sensor diproses menjadi informasi posisi benda oleh pemroses sinyal sensor. Informasi posisi benda kemudian diteruskan ke pengontrol untuk pengambilan keputusan. Sinyal-sinyal komando yang diberikan pada aktuator untuk menggerakkan benda menuju sasaran dihasilkan oleh bagian pengontrol sistem docking. Jadi, ketepatan informasi posisi benda ikut menentukan ketepatan gerak benda untuk melakukan docking.

Mata manusia sebagai sensor, informasi yang diperoleh dari sasaran dipetakan menjadi sebuah informasi yang dimengerti oleh otak. Secara matematis pemetaan tersebut sangat sulit untuk dilakukan karena variabel yang diukur oleh mata tersebut sangat banyak. Masalah ini juga akan kita peroleh dalam mencari hubungan pemetaan pola masukan yang merupakan kombinasi besar hasil pengukuran dari sejumlah

sensor menjadi informasi keluaran yang dapat dimengerti oleh pengontrol utama sistem **docking**. Sangat diperlukan suatu cara yang lebih mudah untuk mendapatkan fungsi sistem yang dapat memetakan pola masukan menjadi pola keluaran yang diharapkan.

**Jaringan Syaraf Tiruan ( JST )**, adalah suatu model proses komputasi berupa susunan sel-sel syaraf tiruan, mirip dengan apa yang diproses oleh sel-sel syaraf biologis. JST memiliki sifat adaptif, **fault-tolerance** dan mampu belajar. JST dapat diajari untuk menjadi sebuah sistem **(black-box)** yang memetakan pola masukan menjadi keluaran yang diinginkan. Sifat mampu belajar dari JST merupakan salah satu solusi yang dapat dimanfaatkan untuk memperoleh fungsi alih sebuah sistem tanpa harus mencari hubungan matematis antara masukan dan keluarannya.

Begitu selesai dilatih, respon sebuah JST dapat menjadi tidak peka terhadap perubahan kecil di masukannya sampai suatu tingkat tertentu. Kemampuan untuk mengatasi derau dan distorsi agar mampu melihat pola sebenarnya merupakan hal yang sangat penting dalam pengenalan pola lingkungan sehari-hari yang sesungguhnya.

JST juga cocok untuk dimanfaatkan dalam sistem docking sebagai pemroses sinyal sensor, karena JST memiliki sifat mampu belajar, adaptif dan kemampuan untuk mengenal pola.

## 1.2 Rumusan dan Batasan Masalah

Tujuan Penelitian Tugas Akhir ini adalah mempelajari pemanfaatan sifat belajar JST dan penerapannya dalam sistem docking. Akan dibuat sebuah simulator JST dengan menggunakan komputer. JST yang digunakan adalah model umpan maju (**feed forward**) dengan kaidah belajar propagasi balik ( **Back Propagation** ) dan disimulasikan dengan sebuah komputer IBM PC.

Sasaran yang ingin dicapai dari tugas akhir ini adalah untuk mengetahui konsep jaringan syaraf tiruan **feed forward** dan penerapannya dalam sistem docking. Konsep penerapan yang dimaksudkan adalah mengungkapkan parameter-parameter JST yang berpengaruh dalam penerapan tersebut serta pengaruhnya dalam menentukan kinerja sistem penentuan posisi.

JST diajari untuk membentuk suatu fungsi yang dapat memetakan pola-pola masukan sensor menjadi informasi benda. Penelitian ini dibatasi hanya pada masalah pembuatan paket belajar dan proses belajar JST untuk membentuk sebuah fungsi sistem yang dapat memetakan masukan dari sensor menjadi informasi posisi benda, tidak akan membahas sistem docking secara keseluruhan.

Untuk keperluan simulasi dibutuhkan sebuah **ADC 0808** yang sudah direalisasikan dalam sebuah Card di IBM PC. ADC tersebut akan melakukan konversi sinyal analog dari sensor menjadi sinyal digital. Kedelapan keluaran ADC diambil sebagai lapisan masukan. Simulasi ini menggunakan bahasa pemrograman

Pascal ( **Turbo Pascal 7.0** ). Sensor yang dipergunakan dalam penelitian ini adalah menggunakan **photo-transistor** untuk mengukur besarnya sinar infra merah yang dipancarkan oleh target.

Untuk mencapai sasaran yang diinginkan tersebut, maka penelitian ini dilakukan urutan metoda sebagai berikut:

1. Studi literatur

   Studi literatur dilakukan untuk mengetahui konsep dasar JST-BPN, sistem docking secara umum serta sistem penentuan posisi.

2. Pembuatan Model Simulator

   Model yang akan dibuat adalah model dari sistem sistem sensor penentuan posisi serta simulator JST.

3. Percobaan

   Percobaan dilakukan bertujuan untuk mendapatkan data-dat yang diperlukan untuk mengetahui pengaruh parameter pada kinerja sistem.

4. Analisis

## 1.3 Sistematika Pembahasan

Laporan penelitian Tugas Akhir ini disusun dalam enam bab, masing-masing bab berisi:

**Bab I Pendahuluan**

Dalam bab ini dijelaskan secara ringkas latar belakang masalah, perumusan masalah dan batasan masalah penelitian.

**Bab II Jaringan Syaraf Tiruan**

Bab ini akan membahas susunan serta algoritma belajar Jaringan Syaraf Tiruan yang digunakan dalam penelitian ini. Pada awal bab akan dibahas pemodelan sel dan susunan sel yang dipakai. Selanjutnya akan diuraikan algoritma **Back Propagation.**

**Bab III Pemrosesan Sinyal Sensor Dalam Sistem Docking**

Isi bab ini membahas tentang sistem docking dan pengajuan JST untuk memecahkan suatu masalah dalam sistem docking.

**Bab IV Aplikasi JST Untuk Sistem Penentuan Posisi Target**

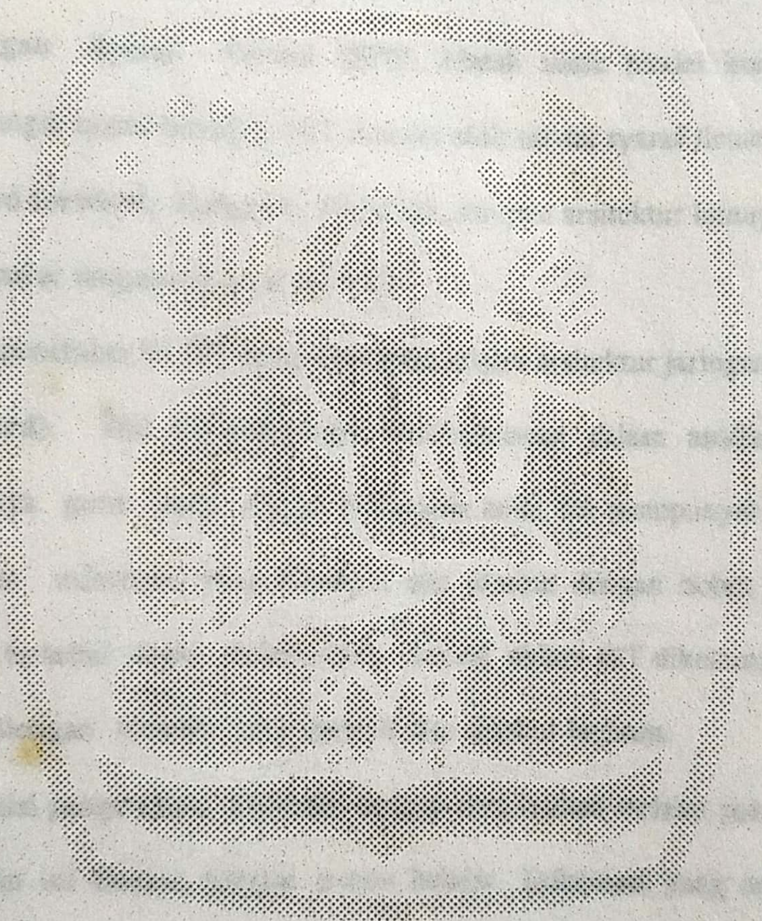
Pada bab ini akan dibahas tentang pembentukan fungsi sistem oleh JST, algoritma belajar pembentukan fungsi serta prosedur pelatihan yang dilakukan untuk membentuk fungsi tersebut.

**Bab V Uji Coba dan Analisa**

Bab ini akan membahas hasil uji coba yang dilakukan, serta pengukuran terhadap parameter dari JST yang berpengaruh dalam pembentukan fungsi.

**Bab VI Kesimpulan dan Tindak Lanjut**

Kesimpulan dari hasil penelitian akan diberikan pada bab ini. Pada akhir bab ini kan diberikan beberapa saran mengenai kemungkinan pengembangan lebih lanjut dari penelitian ini.



# BAB II

# JARINGAN SYARAF TIRUAN

**Jaringan Syaraf Tiruan** (JST), adalah suatu model komputasi yang menirukan fungsi syaraf biologis. JST disusun oleh sel-sel syaraf tiruan berdasarkan arsitektur **feed forward**, **Kohonen**, **Hopfield**, ataupun arsitektur lainnya. Arsitektur JST yang dipakai tergantung pada aplikasi.

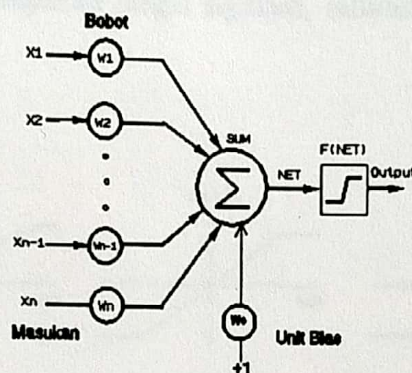
Pada penelitian ini JST yang digunakan adalah arsitektur jaringan umpan maju (**feed forward**). Hal ini mengingat kesederhanaan dalam analisisnya namun memiliki daya guna yang tinggi. Hubungan antar sel mempunyai nilai tertentu sesuai dengan informasi yang disimpan dan disebut dengan bobot interkoneksi. Harga bobot tersebut dapat diubah-ubah. Sel-sel dalam JST dikelompokkan dalam beberapa golongan tertentu yang spesifik dan disebut **lapisan**.

Akuisisi pengetahuan dilakukan dengan cara latihan melalui paket-paket data. Proses latihan ini disebut sebagai proses belajar. Informasi yang telah dipelajari disimpan dalam bentuk bobot-bobot interkoneksi. Sebelum tercapai bobot-bobot interkoneksi yang sesuai dengan galat minimum proses belajar tetap dilakukan dengan

cara mengubah bobot-bobot interkoneksi. JST dapat diimplementasikan dalam bentuk perangkat lunak atau perangkat keras.

## 2.1. Model Sel Syaraf

Model sebuah sel syaraf diperlihatkan pada Gambar 2.1.




**Gambar 2.1 Model Sel Syaraf**

Sebuah sel syaraf akan menjumlahkan semua masukan dari luar dirinya. Masukan ke j, $\mathbf{X}_j$, pada sebuah sel syaraf, akan mempunyai bobot sebesar $\mathbf{W}_j$. Hasil-hasil pembobotan ini kemudian dijumlahkan sebagai keluaran NET yang dihasilkan oleh sel syaraf (Persamaan 2.1.).

$$NET=\sum_{j=1}^{n}(x_j w_j)+w_0 \qquad (2.1)$$

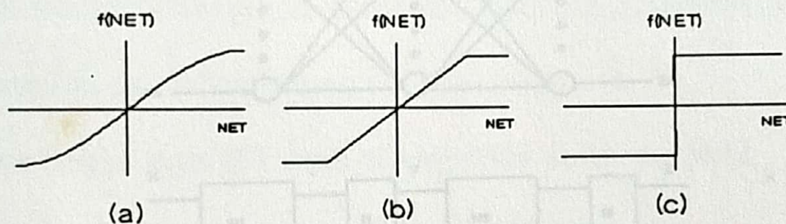
dengan:

j : indeks masukan ke j pada sel syaraf.

Keluaran **NET** dimasukkan ke sebuah fungsi aktifasi untuk memperoleh harga keluaran sel syaraf tersebut.

$$O = f(NET) = f\left(\sum_{j=1}^{n}(x_j w_j) + w_0\right) \qquad (2.2)$$

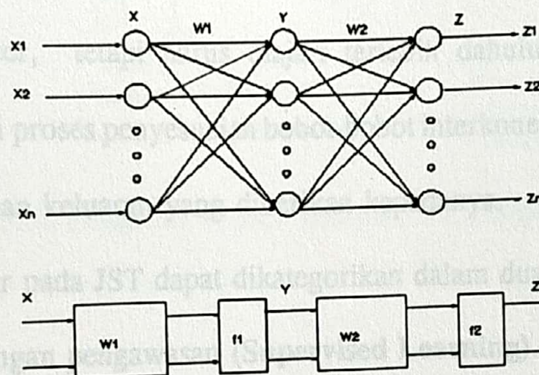
Fungsi aktifasi dapat merupakan fungsi sigmoid, saturasi ataupun linier (Gambar 2.2.).




**Gambar 2.2 Fungsi-fungsi Aktifasi : (a) Sigmoid, (b) Linear, (c) Saturasi.**

## 2.2. Arsitektur Jaringan Umpan Maju (Feed Forward)

Arsitektur jaringan umpan maju (Gambar 2.3) tersusun dari beberapa lapis. Setiap sel syaraf pada suatu lapis hanya dihubungkan dengan sel-sel syaraf pada lapis sebelum atau sesudahnya. Pada lapis yang sama tidak ada hubungan diantara sel-sel syaraf. Arsitektur ini dapat digunakan untuk mengenal pola serta memiliki analisis yang lebih sederhana.




**Gambar 2.3 Jaringan Syaraf Tiruan Feed Forward**

Keluaran JST umpan maju yang tersusun atas tiga lapis, diperlihatkan pada persamaan (2.3).

$$O_{sel} = f_2(W2.Y) \tag{2.3}$$
$$= f_2(W2.f_1(W1.X))$$

Dari arsitektur dan model matematisnya, terlihat bahwa JST umpan maju bersifat statis. Keluaran pada suatu saat hanya ditentukan oleh bobot-bobot interkoneksi dan masukan pada saat itu juga.

## 2.3. Konsep Belajar Dalam JST

Ciri utama JST adalah mempunyai kemampuan untuk diajari. Agar jaringan berfungsi menjadi sebuah sistem yang diinginkan, JST bukan diprogram seperti pada sistem komputer, tetapi harus diajari terlebih dahulu. Belajar pada JST didefinisikan sebagai proses penyesuaian bobot-bobot interkoneksinya, sesuai dengan pasangan masukan dan keluaran yang diberikan kepadanya.

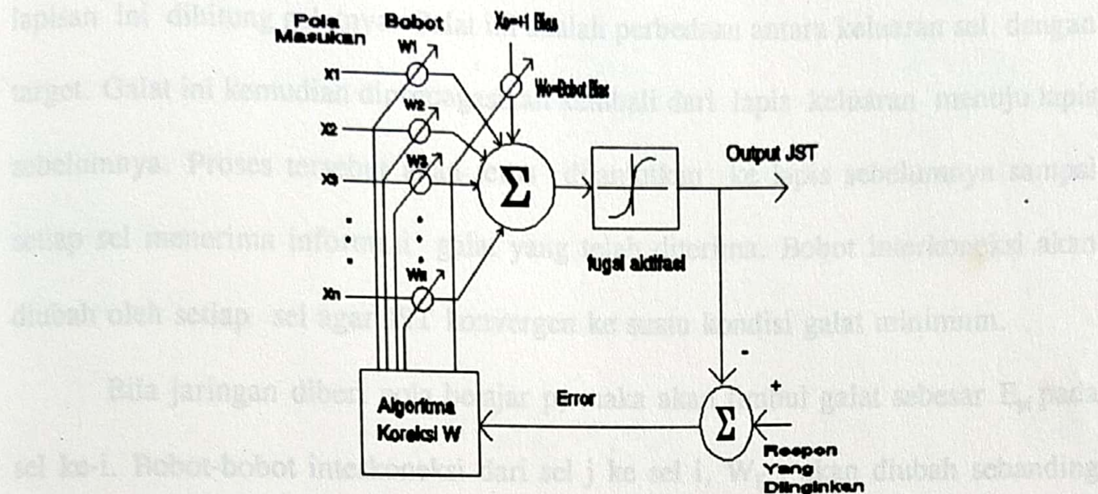
Proses belajar pada JST dapat dikategorikan dalam dua jenis, yaitu:

a. belajar dengan pengawasan (**Supervised Learning**).

b. belajar tanpa pengawasan (**Unsupervised Learning**).

Proses belajar dengan pengawasan memerlukan jawaban keluaran yang diinginkan sebagai dasar pengubahan bobot. Pasangan masukan (soal) dengan keluaran (jawaban) yang diperlukan disebut dengan paket belajar. JST diajari untuk menyelesaikan persoalan-persoalan yang terdapat dalam paket belajarnya. Sedangkan pada proses belajar tanpa pengawasan, JST akan mengubah bobot-bobot interkoneksi dengan sendirinya, sebagai tanggapan atas masukan tanpa memerlukan jawaban targetnya. Dalam penelitian ini proses belajar yang digunakan

adalah proses belajar dengan pengawasan. Diagram blok proses belajar dengan pengawasan dapat direpresentasikan seperti pada Gambar 2.4.




**Gambar 2.4 Blok Diagram Proses Belajar JST dengan pengawasan**

## 2.4. Proses Belajar Pada JST

Kaidah belajar yang dipakai pada JST dengan arsitektur jaringan umpan maju adalah **backpropagation (BP)**. Kaidah ini merupakan hasil modifikasi kaidah penurunan gradien oleh David E. Rummelhart dkk, sehingga dapat digunakan untuk jaringan berlapis banyak.

Dengan kaidah **BP**, maka galat pada lapis keluaran dipropagasikan ke belakang untuk mendapatkan galat lokal sel-sel pada lapisan di belakangnya, sehingga sel-sel pada lapis dalam dapat diajari. Suatu masukan yang diberikan ke jaringan akan dipropagasikan dari sel-sel masukan menuju sel-sel pada lapis selanjutnya

hingga sampai pada sel-sel lapis keluaran. Pada lapis keluaran, keluaran setiap sel dibandingkan dengan keluaran yang dinginkan (target), dan untuk setiap sel pada lapisan ini dihitung galatnya. Galat ini adalah perbedaan antara keluaran sel dengan target. Galat ini kemudian dipropagasikan kembali dari lapis keluaran menuju lapis sebelumnya. Proses tersebut akan terus dilanjutkan ke lapis sebelumnya sampai setiap sel menerima informasi galat yang telah diterima. Bobot interkoneksi akan diubah oleh setiap sel agar JST konvergen ke suatu kondisi galat minimum. .

Bila jaringan diberi pola belajar p, maka akan timbul galat sebesar $E_{pi}$ pada sel ke-i. Bobot-bobot interkoneksi dari sel j ke sel i, $W_{ij}$ , akan diubah sebanding dengan gradien fungsi galat terhadap perubahan bobot. Kecepatan perubahan ini ditentukan oleh harga konstanta laju belajar, $\eta$.

$$\Delta_p W_{ij} = \eta . \frac{\partial E_{pi}}{\partial W_{ij}} \qquad (2.4)$$

Karena galat $E_{pi}$ yang terjadi disebabkan keluaran sel $NET_{pi}$, sedangkan keluaran $NET_{pi}$ disebabkan karena bobot-bobot yang dimilikinya, Wij, sehingga persamaan di atas dapat ditulis kembali dengan menggunakan perubahan potensial aktifasi, $\partial NET_{pi}$.

$$\frac{\partial E_{pi}}{\partial W_{ij}}=\frac{\partial E_{pi}}{\partial NET_{pi}}.\frac{\partial NET_{pi}}{\partial W_{ij}} \qquad (2.5)$$

Dari persamaan (2.5) dapat diketahui bahwa syarat agar persamaan (2.4) dapat dipecahkan adalah fungsi aktifasi sel harus dapat didiferensiasikan (**differentiable**)[1]. Berdasarkan persamaan (2.1), jika $O_{pj}$ sebagai masukan dari lapis ke-j untuk pola p, keluaran $NET_{pi}$ dapat dituliskan seperti pada persamaan (2.6).

$$NET_{pi}=\sum_{j}(W_{ij}.O_{pj}) \qquad (2.6)$$

Dalam setiap tahap penyesuaian bobot, pola p dipertahankan dan masukan sel i ($O_{pj}$) dibuat konstan (Persamaan 2.7). Parameter $\theta_j$ adalah unit bias yang diberikan kepada sel syaraf ke-j. Unit bias, tidak seperti sel yang lain, tidak mempunyai masukan dan memberikan nilai keluaran yang harganya konstan, yaitu 1. Unit bias ini sangat membantu dalam mempercepat proses belajar dan dalam pencapaian titik galat minimum yang diharapkan.

$$\frac{\partial NET_{pi}}{\partial W_{ij}}=O_{ij} \qquad (2.7)$$

Variabel baru yaitu galat lokal sel ke-i untuk pola p, $\delta_{pi}$, didefinisikan seperti pada persamaan (2.8).

$$\delta_{pi} \Delta - \frac{\partial E_{pi}}{\partial NET_{pi}} \quad (2.8)$$

Persamaan (2.4) sebagai persamaan perubahan bobot dalam satu siklus, dapat diubah menjadi persamaan (2.9).

$$\Delta_p W_{ij} = \eta . \delta_{pi} . O_{pj} \quad (2.9)$$

Untuk mencari galat lokal sel, maka definisi persamaan (2.8) harus dimanipulasi. Ruas kanan dari persamaan tersebut, dapat dituliskan seperti pada persamaan (2.10).

$$\frac{\partial E_{pi}}{\partial NET_{pi}} = \frac{\partial E_{pi}}{\partial O_{pi}} . \frac{\partial O_{pi}}{\partial NET_{pi}} \quad (2.10)$$

Sedangkan $O_{pi} = f( NET_{pi} )$.

$$\frac{\partial O_{pi}}{\partial NET_{pi}} = \frac{\partial f(NET_{pi})}{\partial NET_{pi}} \quad (2.11)$$

Pada lapis keluaran, galatnya dapat dicari langsung dari perbedaan keluaran JST, $O_{pi}$, terhadap acuan (target) $Y_{pi}$. Dan $\partial E_{pi}/\partial O_{pi}$ dapat dicari langsung dari galat $E_{pi}$.

$$E_{pi} \triangleq \frac{1}{2}.(Y_{pi}-O_{pi})^2 \qquad (2.12)$$

$$\frac{\partial E_{pi}}{\partial O_{pi}} = -(Y_{pi}-O_{pi}) \qquad (2.13)$$

Dari kedua persamaan di atas, galat lokal sel ke-i pada tiap lapis keluaran pada persamaan (2.8) dituliskan seperti pada persamaan (2.14).

$$\delta_{pi} = (Y_{pi}-O_{pi}) f'(NET_{pi}) \qquad (2.14)$$

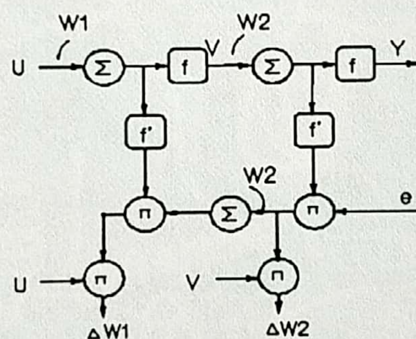
Pada lapis dalam, galat lokal dari semua sel pada lapisan di atasnya harus dipropagasi-balikkan untuk mencari galat lokal pada tiap-tiap sel di lapis tersebut. Pada lapisan ini berlaku persamaan (2.15).

$$\begin{aligned}\frac{\partial E_{pi}}{\partial O_{pi}} &= \sum_k \left(\frac{\partial E_{pi}}{\partial NET_{pk}} . \frac{\partial NET_{pk}}{\partial O_{pi}}\right) \\ &= \sum_k (-\delta_{pk} W_{ki})\end{aligned} \qquad (2.15)$$

Sepanjang lapis k dilakukan penjumlahan, mulai dari lapis keluaran hingga lapis yang akan diajar. Galat lokal dari setiap sel di lapis keluaran dipropagasibalikkan sampai ke sel yang dituju. Jadi, galat lokal sel pada lapis di bawah menjadi:

$$\delta_{pi} = f'(NET_{pi}) \sum_{k} (\delta_{pk} W_{ki}) \qquad (2.16)$$

Algoritma jaringan propagasi balik dapat digambarkan secara diagramatik, seperti pada Gambar 2.5.




**Gambar 2.5 Model Diagramatik Jaringan Propagasi Balik**

# BAB III

# PEMROSESAN SINYAL SENSOR DALAM SISTEM DOCKING

**Sistem docking**, adalah suatu sistem yang berfungsi untuk menempatkan suatu benda ke tempatnya. Dalam melakukan proses docking, sistem docking harus mengetahui posisi benda terhadap tempat sasaran (target). Informasi tersebut diambil dari hasil pengukuran oleh sensor-sensor yang diproses menjadi informasi posisi target.

Dalam mengambil data dari lingkungan dengan sensor, sinyal hasil pengukuran tersebut mengandung noise (derau). Hasil pengukuran yang mengandung noise tersebut akan memberikan informasi yang salah kepada sistem, sehingga keputusan yang diambil oleh sistem akan menjadi salah.

## 3.1. Pemrosesan Sinyal Sensor Untuk Menentukan Posisi Target

Secara umum sistem penentuan posisi dapat digambarkan seperti pada Gambar 3.1.

# BAB III

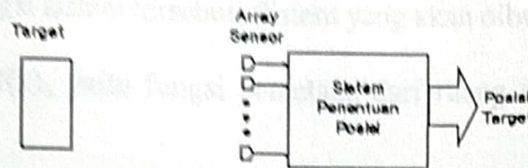
# PEMROSESAN SINYAL SENSOR DALAM SISTEM DOCKING

**Sistem docking**, adalah suatu sistem yang berfungsi untuk menempatkan suatu benda ke tempatnya. Dalam melakukan proses **docking**, sistem docking harus mengetahui posisi benda terhadap tempat tujuan (target). Informasi tersebut diambil dari hasil pengukuran oleh **array-sensor**, yang diproses menjadi informasi posisi target.

Dalam mengambil data dari lingkungan dengan sensor, sinyal hasil pengukuran tersebut mengandung **noise** (derau). Hasil pengukuran yang mengandung noise tersebut akan memberikan informasi yang salah kepada sistem, sehingga keputusan yang diambil oleh sistem akan menjadi salah.

## 3.1. Pemrosesan Sinyal Sensor Untuk Menentukan Posisi Target

Secara umum sistem penentuan posisi dapat digambarkan seperti pada Gambar 3.1.




**Gambar 3.1 Sistem Penentuan Posisi**

Mahluk hidup memiliki kemampuan mengenali objek ataupun peta lingkungan melalui polanya, walaupun pola tersebut pada suatu saat yang lain memiliki derau yang sangat tinggi. Kesulitan yang terdapat dalam sistem pemroses sinyal sensor diantaranya adalah :

1. Adanya derau dari lingkungan yang berubah-ubah;
2. Perbedaan karakteristik dari tiap-tiap sensor yang digunakan;
3. Jumlah sensor yang banyak;

Kesulitan di atas akan menyebabkan kesulitan pada saat membuat model matematis dari pengolah sensor. Pengolah sinyal dalam sistem docking haruslah bersifat adaptif dan mampu mengatasi noise. Sistem pengolah sinyal seperti ini diharapkan mampu mengolah pola sinyal sensor tersebut dengan tepat memetakannya menjadi informasi estimasi posisi target. Sistem yang mampu belajar merupakan salah satu solusi yang menarik.

Pemroses sinyal sensor merupakan sebuah fungsi sistem yang memetakan pola-pola masukan dari array sensor menjadi informasi keluaran. Permasalahan utama

dalam pemroses sinyal adalah mendapatkan model matematis yang dapat merepresentasikan fungsi sistem tersebut. Sistem yang akan dibentuk dapat dinyatakan dengan suatu fungsi, f(x), yaitu fungsi pemetaan dari ruang masukan X ke dalam ruang keluaran Y.

Menurut teorema Wierstrass, suatu fungsi kontinu, f(x), pada interval tertutup [a,b] dapat didekati oleh sebuah polinomial, P(x), dengan batas error $\epsilon$ yang diperbolehkan pada interval tersebut. Secara matematis dapat dituliskan pada persamaan (3.1).

$$\|f(x)-P(x)\| \leq \epsilon \qquad (3.1)$$

Sistem yang linier dan tidak linier dapat didekati oleh polinomial Kolmogorov - Garbor yang dapat dinyatakan dengan persamaan (3.2).

$$f(x_1,x_2,...,x_n)=w_0+\sum_i w_i x_i+\sum_i \sum_i w_{ij} x_i x_j+... \qquad (3.2)$$

Ketelitian dari model matematis yang diperoleh tergantung kepada jumlah suku di dalam polonomial pada persamaan (3.2).

Dengan menggunakan sistem yang mampu belajar fungsi, f(x), dapat didekati oleh fungsi, P(x). Selama proses belajar sistem akan mencari fungsi polinomial, P(x), yang merupakan hubungan pasangan masukan dan keluaran.

Salah satu sistem yang mampu belajar adalah Jaringan Syaraf Tiruan, dan dapat dimanfaatkan sebagai alternatif pembentukan fungsi. Salah satu solusi yang sangat baik untuk pemroses sinyal sistem docking tersebut adalah dengan menggunakan JST. JST mempunyai karakteristik utama mampu belajar, adaptif dan **fault tolerance.**

### 3.2 Jaringan Syaraf Tiruan Sebagai Sistem Belajar

Jaringan Syaraf Tiruan merupakan salah satu sistem yang mampu belajar, pada saat belajar Jaringan Syaraf Tiruan akan mencari hubungan antara masukan dan keluaran dengan mengatur parameter-parameternya dalam hal ini bobot-bobot interkoneksi antara sel syaraf.

Kemampuan JST untuk belajar merupakan solusi bagi sulitnya mencari model matematis dari hubungan keluaran dan masukan suatu sistem ataupun pemetaan. JST dapat diajari untuk membentuk suatu fungsi, yaitu dengan memberikan paket belajar. Paket belajar disini didefinisikan sebagai pasangan soal (masukan) dan jawaban (target) yang diberikan selama proses belajar. Selama proses belajar JST akan mencari sendiri hubungan pemetaan antara keluaran terhadap masukannya. Dengan kata lain, kita tidak perlu lagi mencari hubungan matematis antara masukan dan keluaran suatu sistem untuk memperoleh fungsi yang diinginkan.

Sedangkan sifat JST yang **fault tolerance**, akan menghasilkan sifat kebal

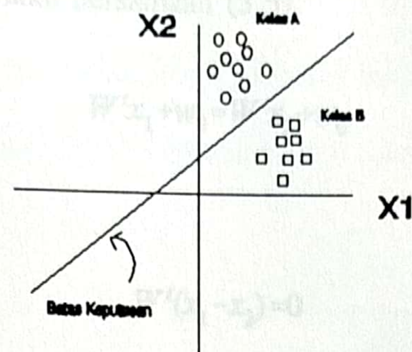
terhadap kesalahan. Suatu sistem dikatakan bersifat kebal terhadap kesalahan jika sistem tersebut masih mampu menangani suatu permasalahan walaupun mengalami kerusakan pada salah satu atau beberapa komponen sistem. Karena JST berarsitektur yang paralel masif, maka jika ada sebagian sel yang mati pada lapisan dalam (bukan pada layer input dan output) beberapa sel lain yang masih berfungsi akan tetap bekerja menyesuaikan parameter agar fungsi JST keseluruhan dapat dijaga. Proses pembentukan fungsi yang diinginkan masih dapat berlangsung, walaupun dengan ketelitian yang lebih rendah.

### 3.3 Pembentukan Fungsi Pemetaan Dengan JST

**Jaringan Syaraf Tiruan** dalam melakukan pembentukan fungsi pemetaan adalah membagi ruang pola masukan ke dalam kelas-kelas masukan yang diinginkan yang mencerminkan suatu daerah keputusan. Selama proses belajar JST akan membentuk batas-batas keputusan, sehingga keseluruhan batas keputusan tersebut membentuk suatu hyperplanar seperti pada Gambar 3.2. Semakin banyak hyperplanar yang dibentuk, maka daerah keputusan yang dibentuk akan semakin banyak.

Bentuk batas keputusan tersebut sangat beragam, tetapi untuk menyederhanakan masalah hanya akan membahas batas keputusan linier. Hyperplanar, secara matematis dapat ditulis dengan persamaan (3.3).

$$w_0+w_1x_1+w_2x_2+\ldots+w_nx_n=G(X) \quad (3.3)$$




**Gambar 3.2 Pengklassifikasian Pola.**

dengan $w_0, w_1, w_2, .., w_n$ adalah parameter pembentuk fungsi, dan $x_1, x_2, .., x_n$ adalah variabel kordinat titik-titik pembentuk hyperplane. Persamaan (3.3) disebut juga dengan fungsi diskriminan linier ( **Linear Discrimination Function** ). Jika didefinisikan suatu vektor **X** dengan, $\mathbf{X} = (x_1, x_2, .., x_n)$, dan vektor **W** dengan, $\mathbf{W} = (w_1, w_2, .., w_n)$, maka persamaan (3.3) dapat ditulis kembali dengan persamaan (3.4).

$$w_0 + \langle W, X \rangle = G(X) \qquad \textbf{(3.4)}$$

G(X) dapat dipandang sebagai jarak untuk masing-masing titik $(x_1, x_2)$.

Bila persamaan (3.3) dikaitkan dengan Jaringan Syaraf Tiruan, maka **X** adalah vektor pola masukan dan **W** adalah vektor bobot yang dicari selama proses belajar. Jika komponen vektor $x_1$ dan $x_2$ keduanya terletak pada daerah batas

keputusan, maka akan berlaku persamaan (3.5).

$$W^t x_1 + w_0 = W^t x_2 + w_0$$

atau

$$W^t (x_1 - x_2) = 0 \quad (3.5)$$

dari persamaan ini berarti w normal terhadap hyperplane.

Jadi, sifat geometri dari hyperplanar adalah :

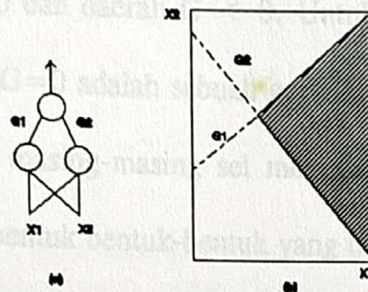
1. Hyperplanar dan vektor bobot w saling tegak lurus,
2. Jarak suatu titik S dengan hyperplanar adalah seperti persamaan (3.6).

$$D(s) = \left| \frac{w_0 + <W,S>}{|W|} \right| \quad (3.6)$$

Secara umum fungsi G(X) membagi ruang ciri-ciri yang dimiliki dengan suatu hyperplanar yang arahnya (orientasi) ditentukan oleh vektor normal W dan lokasinya ditentukan oleh $w_0$.

Kemampuan Jaringan Syaraf Tiruan untuk membentuk daerah-daerah keputusan sangat tergantung kepada jumlah sel dan jumlah lapisan. Jaringan Syaraf Tiruan dengan satu lapis mampu membagi daerah keputusan menjadi dua bagian yang dipisahkan oleh hyperplanar linier. Pada Jaringan Syaraf Tiruan dua lapis, sel-sel

pada lapisan kedua merepresentasikan interseksi tiap-tiap setengah ruang yang dipisahkan masing-masing hyperplane yang dibentuk pada lapisan pertama. Interseksi tersebut membentuk suatu daerah cembung dalam ruang masukan X, seperti pada Gambar 3.3.



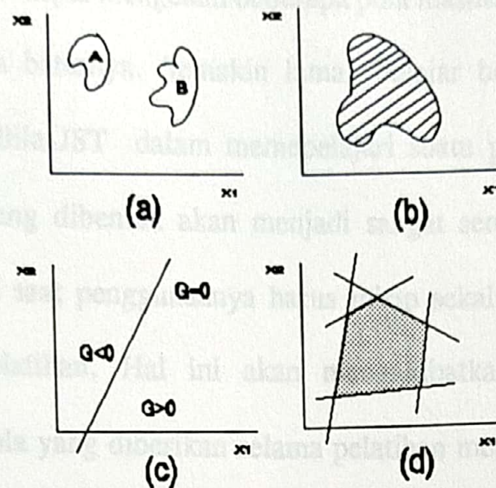
**Gambar (3.3) Hubungan Antar Lapis JST.**

Jumlah sisi dari daerah cembung sama dengan jumlah hyperplane yang berinterseksi, dan memberikan indikasi beberapa sel yang dibutuhkan untuk keperluan tertentu pada jaringan dua lapis. Sedangkan Jaringan Syaraf Tiruan dengan tiga lapis akan mampu memisahkan pola dalam ruang masukan menjadi beberapa hypercube, dimana tiap hypercube merupakan hasil dari lapis kedua. Menurut Lippmann (1987), jumlah minimum sel pada layer kedua sama dengan jumalah daerah keputusan yang tidak terhubung[3]. Jaringan Syaraf Tiruan dengan jumlah layer yang lebih bànyak akan dapat mengklassifikasi pola-pola masukan ke dalam hypercube yang lebih kompleks lagi.

Proses pembentukan daerah keputusan oleh Jaringan Syaraf Tiruan tiga lapis dapat diurutkan sebagai berikut :

1. Pada lapisan dalam, masing-masing sel membagi daerah menjadi dua daerah keputusan, sepanjang bidang permukaan G(x) pada $x_1,x_2,..,x_n$ dengan gradien bidang permukaan dibedakan oleh $w_1,w_2,..,w_n$ dan ketinggian garis batas keputusan G=0 adalah W0. Dengan asumsi bahwa gradien adalah tidak nol, maka akan ada aderah $G > 0$ dan daerah $G < 0$. Untuk pola masukan dua dimensi tempat kedudukan titik G=0 adalah sebuah garis seperti pada persamaan (3.3).
2. Pada lapisan keluaran, masing-masing sel menggabungkan satu atau beberapa hyperplane untuk membentuk bentuk-bentuk yang berubah-ubah. Setiap sel pada lapisan keluaran akan menerima masukan dari setiap sel pada lapis dalam, yang mencirikan kedudukan masing-masing bidang. Seluruh informasi tersebut digabungkan sehingga keluaran sel dapat mengkombinasikan hyperplanar untuk membentuk daerah keputusan yang lebih kompleks lagi. Jika jumlah sel di lapisan tersembunyi lebih banyak maka jumlah hyperplane akan lebih banyak sehingga bentuk daerah keputusan yang terbentuk pada lapisan keluaran akan lebih kompleks lagi.

Jadi, ketelitian jaringan ditentukan oleh jumlah sel pada lapisan dalam. Semakin banyak jumlah sel pada lapisan tersembunyi maka daerah keputusan yang terbentuk akan semakin mulus ( **smooth** ). Gambaran proses pembentukan daerah keputusan oleh setiap lapis pada Jaringan Syaraf Tiruan adalah seperti pada Gambar 3.4.



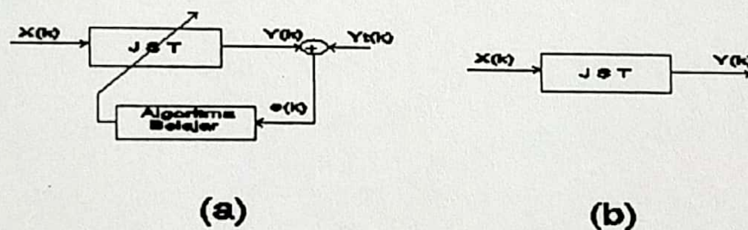

**Gambar 3.4 (a) daerah masukan,(b) daerah keputusan yang diinginkan, (c)daerah keputusan yang dibentuk oleh satu sel, (d) daerah keputusan yang dibentuk pada sel keluaran.**

## 3.4 JST Sebagai Pembentuk Fungsi Pemetaan Sensor

Dengan melihat kemampuan JST untuk belajar, JST dapat dimanfaatkan untuk membentuk suatu fungsi yang diinginkan. Pada saat belajar JST akan digunakan sebagai pengenal pola yang adaptif dengan memasangkan algoritma belajarnya. Selama proses ini JST akan mencari sendiri hubungan antara pola masukan yang diberikan dan pola keluaran yang harus dibangkitkan (target). Pencarian hubungan tersebut dilakukan dengan mengatur bobot-bobot interkoneksi dalam jaringan. Penyesuaian bobot tersebut dilakukan dengan metoda **stepest-descent** melalui iterasi per Epoch sampai batas error yang diizinkan diperoleh. Semakin lama proses pengenalan dilakukan, hasil yang akan dihasilkan akan semakin baik.

Tetapi agar JST dapat mengenali beberapa pola masukan, jumlah iterasi untuk setiap pola harus ada batasnya. Semakin lama belajar belum tentu fungsi yang diinginkan tercapai. Bila JST dalam memepelajari suatu pola terlalu lama, maka daerah keputusan yang dibentuk akan menjadi sangat sempit, sehingga masukan yang diberikan pada saat penggunaanya harus mirip sekali dengan masukan yang diberikan selama pelatihan. Hal ini akan mengakibatkan pola masukan yang berdekatan dengan pola yang diberikan selama pelatihan menjadi tidak dikenalí oleh fungsi sistem yang dibentuk.

Arsitektur pelatihan yang dipergunakan pada sistem docking ini diperlihatkan pada Gambar 3.5. Selama proses belajar JST akan merubah bobot-bobotnya seiring dengan pengurangan error, dan menuruti algoritma propagasi balik.




**Gambar 3.5 (a) Proses belajar pada saat pembentukan fungsi, (b) Penggunaan setelah terbentuk hasil belajar**

Fase pelatihan bagi JST, memakan waktu yang cukup lama karena harus dilakukan iterasi terhadap semua pola paket belajar. Seperti dikemukakan sebelumnya, JST akan membentuk suatu fungsi yang diinginkan. Setelah fase

pelatihan dianggap selesai, JST dapat dipergunakan dalam sistem docking. Fase aplikasi dari JST adalah seperti Gambar 3.5.b. Dalam fase aplikasi sifat adaptif dari JST tersebut akan hilang, tetapi sifat **fault tolerance** masih tetap akan ada[1].

# BAB IV
# APLIKASI JST UNTUK
# SISTEM PENENTUAN POSISI TARGET

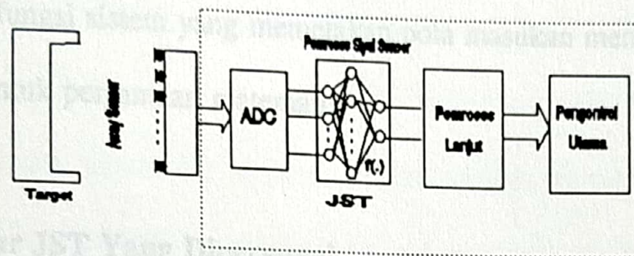
Dalam penelitian ini JST diimplementasikan dengan perangkat lunak yang menggunakan Turbo Pascal 7.0 pada komputer IBM PC. Sebagai sensor yang mengukur jarak benda ke sasaran [illegible] dan penerima infra merah. Jarak benda ke sasaran [illegible] yang diterima kembali oleh sensor. Sinyal dari sensor [illegible] menggunakan ADC 0809 yang direalisasikan dalam [illegible] ADC [illegible] dipasang pada slot perluasan PC. ADC 0809 [illegible] analog, sehingga dapat membaca masukan dari [illegible].

# BAB IV
# APLIKASI JST UNTUK SISTEM PENENTUAN POSISI TARGET

Dalam penelitian ini JST diiplementasikan dengan perangkat lunak yang menggunakan **Turbo Pascal 7.0** pada komputer IBM PC. Sebagai sensor yang mengukur jarak benda ke sasaran dipergunakan pemancar dan penerima infra merah. Jarak benda ke sasaran diukur melalui sinyal pantulan yang diterima kembali oleh sensor. Sinyal dari sensor tersebut dikuantisasi dengan menggunakan ADC 0808 yang direalisasikan dalam sebuah ADC Card. ADC Card tersebut dipasang pada slot perluasan PC. ADC 0808 mempunyai 8 kanal masukan analog, sehingga dapat membaca masukan dari 8 buah sensor secara multipleks.

## 4.1. Sistem Docking

Secara diagram blok sistem **docking** dapat digambarkan seperti pada Gambar 4.1.




**Gambar 4.1 Diagram Blok Sistem Docking.**

Array sensor mengukur besarnya sinyal infra merah yang datang dari sasaran, kemudian diubah ke dalam bentuk sinyal digital oleh **Analog to Digital Converter**. Kemudian diproses secara digital untuk mendapatkan informasi posisi benda terhadap sasaran. Kombinasi dari besarnya pengukuran dari setiap sensor selanjutnya disebut sebagai **pola masukan**.

Pemroses sinyal-sinyal yang datang dari sensor merupakan bagian yang sangat penting, karena bagian tersebutlah yang memetakan pola-pola sinyal masukan menjadi informasi posisi. Bagian pemroses sinyal ini harus mempunyai sifat adaptif dan mampu mengatasi derau yang datang dari lingkungan sekitarnya.

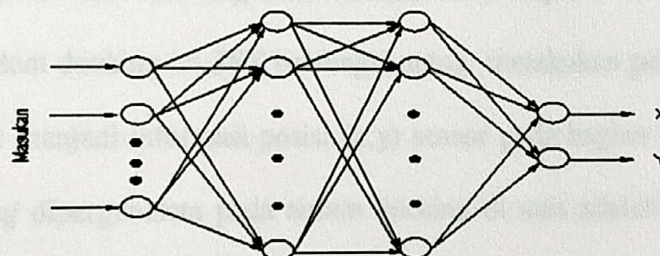
Pola sinyal sensor diartikan sebagai kombinasi angka digital dari hasil pengukuran intensitas sinar infra merah oleh delapan buah **array sensor**. Kedelapan

sinyal output sensor tersebut dapat direpresentasikan menjadi sebuah vektor pola masukan, dan jenis pola tersebut merupakan pola spasial. Dikatakan pola spasial, karena pengukurannya dilakukan secara serentak untuk satu pola.

Pada penelitian ini pemroses sinyal yang digunakan adalah komputer digital serial, sehingga fungsi sistem yang memetakan pola masukan menjadi pola keluaran adalah dalam bentuk persamaan matematis.

## 4.2 Arsitektur JST Yang Dipergunakan

Arsitektur JST yang dipilih untuk sistem docking digambarkan pada Gambar 4.2.




**Gambar 4.2 Arsitektur JST Yang Digunakan Untuk Sistem Docking**

Sinyal dari array sensor dimasukkan ke ADC dan kemudian dikonversikan ke dalam sinyal digital dan dibaca oleh PC sebagai pola masukan. Untuk setiap pola masukan diberikan jawaban melalui keyboard, dan data untuk beberapa posisi (pola masukan) dan jawabannya direkam ke file sebagai paket belajar bagi JST. Pada saat inisialisasi paket belajar dibaca dari file untuk keperluan belajar. JST akan

mempropagasikan data tersebut sampai ke lapisan keluaran.

Selama belajar, JST akan mempropagasikan pola masukan untuk membangkitkan keluaran. Keluaran akan dibandingkan dengan target (keluaran yang diinginkan). Dari pembandingan tersebut akan diperoleh galat, dan galat ini kemudian dipropagasibalikkan untuk mengubah bobot-bobot interkoneksi. Nilai bobot-bobot ini kemudian digunakan kembali untuk membangkitkan keluaran JST bagi pola masukan yang selanjutnya dan proses ini diulang sampai seluruh pola dipropagasikan dan terjadi perubahan bobot-bobot interkoneksi. Proses tersebut terus berlangsung sampai diperoleh batas galat minimum untuk setiap pasangan pola masukan dan keluaran.

### 4.3 Fungsi Pemetaan dan Algoritma Pembentukannya

Pada sistem docking ini **JST** berfungsi untuk melakukan pemetaan dari pola masukan sensor menjadi informasi posisi (x,y) sensor pada bagian keluarannya.

JST yang dipergunakan pada sistem docking di atas adalah jaringan umpan maju ( **feed forward** ) dengan algoritma belajar propagasi balik. Lapisan masukan terdiri dari delapan sel sesuai dengan jumlah sensor yang dipakai. Lapisan keluaran terdiri dari dua sel. Masing - masing sel menghasilkan informasi jarak-x dan jarak-y. Sedangkan pada lapis dalam dapat terdiri dari beberapa sel maupun lapis. Semakin banyak sel dan lapisnya, akan semakin teliti perhitungannya.

Algoritma belajar untuk membentuk fungsi merupakan tahap yang dapat

dirinci sebagai berikut

1. Inisialisasi JST, merupakan tahap penyusunan arsitektur JST yang dipilih. Penyususnan mungkin merupakan tahap awal atau tahap lanjutan, meliputi jumlah sel lapis dalam, konstanta belajar, dan harga bobot-bobot awal sel. Untuk tahap awal belajar, jumlah sel lapis dalam, konstanta belajar ditentukan, dan bobot-bobot interkoneksi diset secara acak antara -0.5 dan 0.5. Dan untuk proses belajar lanjutan, konfigurasi akan mengikuti konfigurasi pada proses sebelumnya dengan konstanta belajar yang dapat ditentukan kemudian.

2. Pola-pola paket belajar yang dibaca dari sensor dibaca kembali untuk keperluan belajar bagi JST. Setiap sel masukan menerima besar harga kuantisasi sinyal satu buah sensor. Kedelapan harga masukan pada lapisan masukan dipropagasikan menuju lapis dalam. Pada setiap sel di lapis tersembunyi seluruh masukan $\mathbf{X_i}$ dijumlahkan, setelah mendapat pembobotan $\mathbf{W_{ji}}$, yang menghasilkan keluaran di setiap sel, $\mathbf{NET_j}$ .

$$NET_j=\sum_{i=1}^{M} X_i.W_{ij} \qquad \textbf{(4.1)}$$

Setelah dilakukan penjumlahan semua sel lapis masukan, hasilnya dimasukkan ke fungsi aktifasi untuk memperoleh keluaran lapis dalam, $\mathbf{O_j}$.

Propagasi dilanjutkan menuju lapis keluaran. Pada lapis ini masukan dari lapis

$$O_j = \frac{1}{1+\exp(-\sum_{i=1}^{M} X_i . W_{ij})} \qquad (4.2)$$

sebelumnya, $\mathbf{O}_j$, akan mendapatkan pembobotan $\mathbf{W}_{ij}$.

$$NET_k = \sum_{j=1}^{J} O_j . W_{kj} \qquad (4.3)$$

3. Keluaran JST akan diperoleh, kemudian akan dibandingkan dengan jarak-x dan jarak-y sebenarnya, yang dimasukkan melalui keyboard.
4. Galat lokal pada lapis keluaran dapat dihitung langsung dari perbedaan antara keluaran jaringan dengan target yang diberikan. Galat lokal sel pada lapis keluaran dapat dihitung dengan persamaan (4.4).

$$\delta_k = (Y_k - O_k) . f'(NET_k) \qquad (4.4)$$

Kemudian dilakukan penyesuaian bobot pada lapis output dengan

konstanta belajar, $\eta$.

$$W_{kj}(k+1) = \eta . \delta_k . O_j + W_{kj}(k) + \beta \Delta W(k-1) \qquad (4.5)$$

Galat lokal $\delta$ pada lapis keluaran ini dipropagasibalikkan menuju lapis dalam untuk menghitung galat pada lapis dalam.

Dengan menggunakan galat lokal $\delta$ pada persamaan (4.6) dilakukan

$$\delta_j = \left[\sum_{k=1}^{K} \delta_k . W_{kj}\right] . f'(NET_j) \quad (4.6)$$

penyesuaian bobot-bobot interkoneksi pada lapis dalam dengan konstanta belajar, $\eta$, dan momentum, $\beta$.

$$W_{ji}(k+1) = \eta . \delta_j . O_j + W_{ji}(k) + \beta \Delta W(k-1) \quad (4.7)$$

6. Langkah 2 sampai dengan 5 diulangi terus sampai diperoleh galat minimum yang diperbolehkan.

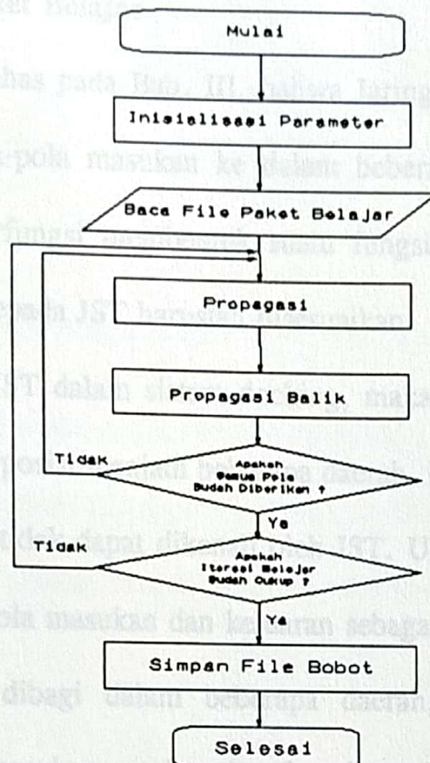
7. Semua parameter JST, yaitu konfigurasi dan bobot -bobot disimpan untuk digunakan dalam proses belajar JST bagi pola masukan yang selanjutnya.

Keseluruhan algoritma proses belajar sistem docking dapat direpresentasikan dengan sebuah diagram alir pada Gambar 4.3.

## 4.4 Prosedur Pelatihan

Pada awal proses, bobot - bobot interkoneksi ditentukan secara acak dan berada pada interval [-0.5 , 0.5]. Harga konstanta belajar dan momentum dipilih bersesuaian dengan jumlah sel lapis dalam dengan melihat tanggapan sistem yang terjadi. Konstanta belajar dapat dipilih dengan melihat tanggapan sinyal JST untuk menuju nilai yang diinginkan. Konstanta belajar yang baik untuk JST biasanya berkisar antara 0.05 sampai dengan 0.25 dan momentum biasanya dipilih sekitar 0.9 ($0 < \beta < 1$).

Jika konstanta belajar sudah dipilih, maka proses belajar dapat dimulai. Bobot-




**Gambar 4.3 Algoritma Belajar JST Untuk Sistem Docking**

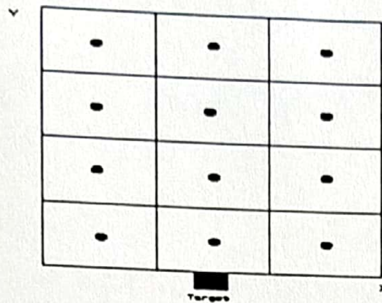
bobot interkoneksi dari hasil pelajaran dari pola pertama disimpan untuk kemudian dipergunakan sebagai bobot awal bagi pola yang kedua, kemudian bobot hasil belajar dari pola yang kedua disimpan lagi untuk dipergunakan sebagai bobot awal bagi pola yang ketiga. Begitu seterusnya sampai semua pola diajarkan kepada Jaringan Syaraf Tiruan tersebut. Sehingga setiap iterasi kita memberikan semua pola untuk pengubahan bobot-bobot interkoneksi JST, dengan kata lain iterasi dilakukan per Epoch.

## 4.5 Representasi Paket Belajar

Seperti telah dibahas pada Bab. III, bahwa Jaringan Syaraf Tiruan bersifat mengklassifikasikan pola-pola masukan ke dalam beberapa kelas. Agar Jaringan Syaraf Tiruan dapat berfungsi membentuk suatu fungsi, maka representasi pola belajar yang diberikan kepada JST haruslah disesuaikan.

Pada penerapan JST dalam sistem docking, maka JST setelah belajar akan membagi seluruh daerah posisi menjadi beberapa daerah. Hal ini akan menyebabkan ada posisi tertentu yang tidak dapat dikenali oleh JST. Untuk itu perlu diteliti cara pengambilan data pola-pola masukan dan keluaran sebagai paket belajar bagi JST.

Seluruh bidang dibagi dalam beberapa daerah, dan berikut kontribusi pengambilan pola-pola masukan untuk setiap daerah terdistribusi secara uniform di bidang dua dimensi dan berada di depan sasaran. Pengambilan pola belajar untuk pola-pola masukan yang uniform dilakukan seperti Gambar 4.4. JST diajari untuk memberi hanya satu jawaban untuk satu kelas pola yang diberikan, sehingga resolusi akan sangat tergantung kepada jumlah kelas. Semakin banyak daerah yang dibuat maka resolusi akan semakin baik, dimana setiap daerah tersebut dibatasi oleh garis-garis batas keputusan.




**Gambar 4.4 Pengambilan pola-pola paket belajar.**

Agar JST mampu membentuk daerah keputusan yang banyak, maka jumlah sel dalamnya harus disesuaikan dan dipilih berapa kira-kira yang optimalnya. Ukuran yang optimal tersebut dilakukan dengan cara coba-coba.

# BAB V

# UJI COBA DAN ANALISA

Pemilihan parameter [illegible]

sangat berpengaruh [illegible]

coba simulasi [illegible] untuk mendapatkan [illegible]

fungsi yang [illegible] melalui proses [illegible]

tersebut terhadap [illegible]

## 5.1 Pengaruh Para[illegible]

**Fungsi**

Tujuan memilih [illegible] adalah [illegible]

tersebut sehingga tidak [illegible]

belajar yang relatif [illegible]

juga akan [illegible]

simulasi ini diambil [illegible] karena akan [illegible]

### 5.1.1. Konstanta Laju Belajar dan Momentum

Konstanta belajar dan momentum akan sangat berpengaruh [illegible]

# BAB V

# UJI COBA DAN ANALISA

Pemilihan parameter Jaringan Syaraf Tiruan dan representasi pola belajar sangat berpengaruh pada terbentuk atau tidaknya fungsi yang diinginkan. Tujuan uji coba simulasi ini adalah untuk mendapatkan struktur JST yang mampu membentuk fungsi yang diinginkan melalui proses belajar. Dan juga untuk melihat respon JST tersebut terhadap pola-pola masukan yang diberikan setelah selesai belajar.

## 5.1 Pengaruh Parameter Jaringan Syaraf Tiruan pada Proses Pembentukan Fungsi

Tujuan memilih parameter JST adalah untuk mengoptimalkan pemakaian JST tersebut sehingga tidak boros dalam pemakaian jumlah sel dan menghabiskan waktu belajar yang relatif singkat. Penggunaan konstanta belajar dan momentum yang tepat juga akan membantu pembentukan fungsi yang mulus. Sedangkan unit bias dalam simulasi ini diambil +1, karena akan sangat membantu dalam proses belajar[2].

### 5.1.1 Konstanta Laju Belajar dan Momentum

Konstanta belajar dan momentum akan sangat berpengaruh kepada kecepatan

belajar. Konstanta belajar yang terlalu besar akan menyebabkan JST tidak dapat membentuk fungsi yang diinginkan, karena step pengubahan bobot yang dilakukan terlalu besar. Memang dengan menggunakan konstanta belajar yang besar akan mempercepat pengurangan galat, tetapi dapat menyebabkan JST tidak mencapai galat minimum. Dengan perkataan lain JST akan terjebak ke dalam minimum lokal, dimana galat terkecil yang diharapkan tidak tercapai.

Salah satu cara untuk mempercepat pencapaian galat minimum tersebut adalah dengan menggunakan momentum. Dengan menggunakan momentum maka konstanta belajar dapat dibuat kecil, sehingga daerah keputusan yang dibentuk akan lebih mulus (**smooth**).

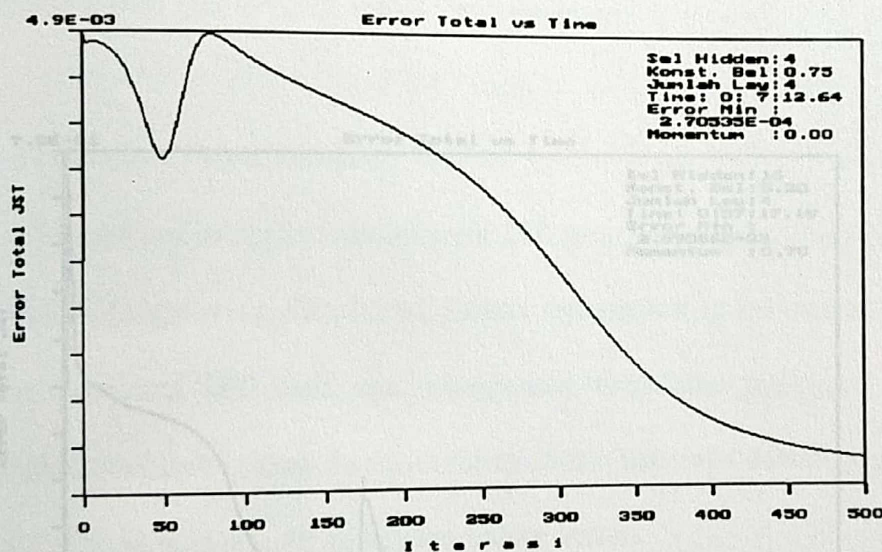
Dari Gambar 5.1 dan Gambar 5.2 dapat dilihat bahwa penggunaan momentum menyebabkan proses belajar yang lebih cepat. Dengan jumlah sel tersembunyi yang sama, proses belajar JST dengan menggunakan momentum akan lebih cepat mencapai galat yang diinginkan daripada proses belajar tanpa menggunakan momentum.

**5.1.2 Jumlah Sel Syaraf**

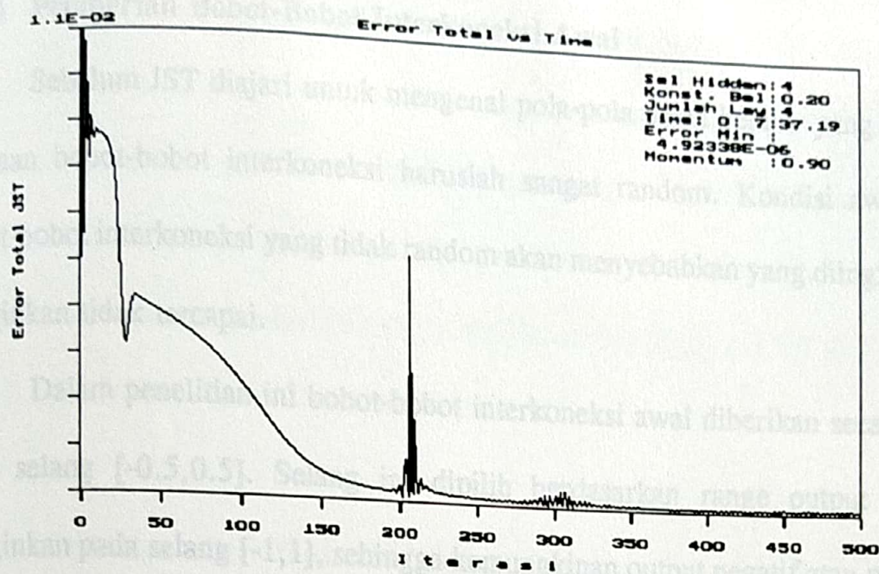
Seperti dikatakan pada bab IV bahwa jumlah sel pada layer tersembunyi sebaiknya sebanyak jumlah daerah keputusan yang tidak terhubung. Penggunaan jumlah sel yang terlalu banyak hanya akan memperlambat proses perhitungan dan pemborosan terhadap pemakaian memori. Gambar 5.3 menunjukkan bahwa

penggunaan sel yang lebih banyak akan menyebabkan waktu untuk satu iterasi bertambah lama.

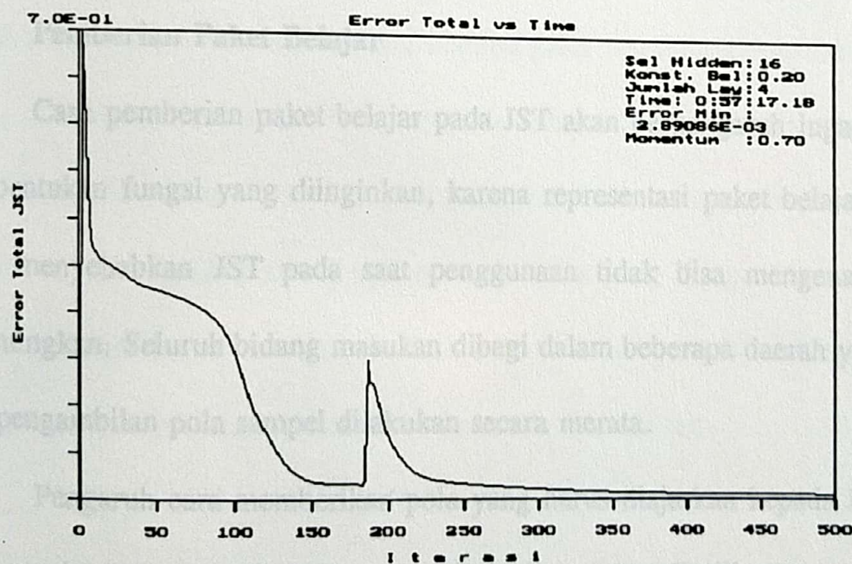
Untuk memilih jumlah sel dapat dilakukan dengan mengambil jumlah sel hidden minimal sebanyak jumlah daerah keputusan yang tidak terhubung. Tetapi dengan itu saja tidak cukup dan harus dilakukan coba-coba sampai JST dapat belajar lebih cepat dan setelah penggunaannya dapat memberi respon yang diinginkan terhadap pola-pola masukan yang diberikan. Setelah melalui uji coba dengan jumlah layer 4, jumlah sel lapisan hidden pertama dan jumlah sel hidden kedua masing-masing sebanyak 16 dan 6. Sedangkan pola belajarnya adalah 15.




**Gambar 5.1 Kurva penurunan galat kuadrat rata-rata total JST pada proses belajar tanpa menggunakan momentum.**




**Gambar 5.2 Kurva penurunan galat kuadrat rata-rata total JST pada proses belajar dengan menggunakan momentum.**




**Gambar 5.3 Kurva galat JST dengan menggunakan 4 lapis, dan lapisan tersembunyi mempunyai 16 sel.**

### 5.1.3 Pemberian Bobot-Bobot Interkoneksi Awal

Sebelum JST diajari untuk mengenal pola-pola paket belajar yang diberikan, keadaan bobot-bobot interkoneksi haruslah sangat random. Kondisi awal dengan bobot-bobot interkoneksi yang tidak random akan menyebabkan yang diinginkan tidak diinginkan tidak tercapai.

Dalam penelitian ini bobot-bobot interkoneksi awal diberikan secara random pada selang [-0.5,0.5]. Selang ini dipilih berdasarkan range output JST yang diinginkan pada selang [-1,1], sehingga kemungkinan output negatif atau positif pada kondisi awal sama. Kondisi ini juga akan ikut mengurangi jumlah iterasi pengubahan bobot-bobot untuk memperoleh bobot yang optimum.

## 5.2 Pemberian Paket Belajar

Cara pemberian paket belajar pada JST akan berpengaruh juga pada proses pembentukan fungsi yang diinginkan, karena representasi paket belajar yang salah akan menyebabkan JST pada saat penggunaan tidak bisa mengenal pola yang dipentingkan. Seluruh bidang masukan dibagi dalam beberapa daerah yang berbeda, dan pengambilan pola sampel dilakukan secara merata.

Pengaruh cara memberikan pola yang harus diajarkan kepada JST terutama pada pembentukan daerah keputusan. Bila pola tersebut disajikan secara acak, maka bentuk hypercube yang harus dibentuk oleh JST akan semakin rumit. Sedangkan bila

pola tersebut diambil dengan distribusi yang merata maka jarak pembatas daerah keputusan ke titik posisi pengambilan pola akan sama untuk setiap pola. Dalam penenlitian ini dipergunakan 12 pola belajar yang mewakili 12 posisi (x,y) yang berbeda dari sensor di depan target dan diambil dengan distribusi merata. Posisi dari kesembilan tersebut dipilih titik (0,3), (-5,3), (5,3), (0,9), (-5,9), (5,9), (0,15), (-5,15), (5,15), (0,21), (-5,21), dan (5,21).

### 5.3 Penentuan Saat Berhenti Belajar

Untuk menentukan kapan berhenti belajar bagi JST dilakukan dengan metoda coba-coba dan sangat tergantung kepada seberapa kecil galat yang diinginkan. Dari hasil pengujian dapat dibandingkan dua matriks bobot yang diperoleh dari dua batas galat yang berbeda. Pada Tabel 5.1 diambil galat maksimum di setiap keluaran JST pada orde satu per seribu, sedangkan pada Tabel 5.2 diambil galat maksimum pada orde satu per seratus ribu. Ternyata galat maksimum dengan orde satu per seribu sudah cukup bagi JST untuk memperoleh bobot yang diinginkan, dan jika iterasi tersebut diteruskan akan menyebabkan pola masukan yang berdekatan dengan pola sampel yang diberikan pada saat belajar tidak dikenali lagi.

Jadi, proses belajar bagi JST ada batasnya dan belajar yang terlalu banyak akan mengakibatkan JST menyimpang dari fungsi yang diinginkan.

## 5.4 Analisa Sifat Mampu Belajar JST dan Kemampuan Mengatasi Derau

Dari Tabel 5.1 dan Tabel 5.2 dapat dilihat bahwa JST dapat diajari untuk membentuk suatu fungsi dengan cara mencari hubungan antara masukan dan keluaran. Pada aplikasi untuk sistem **docking**, JST tidak dapat melakukan ekstrapolasi. Hal tersebut dapat dilihat pada Tabel 5.1 dan Tabel 5.2, jika diberikan pola yang jauh dari pola sampel belajar maka JST akan memberi jawaban yang salah.

Jadi, supaya JST tersebut dapat lebih banyak lagi mengenali pola yang diinginkan haruslah jumlah kelas pola sampel belajar yang diberikan lebih banyak lagi. Semakin banyak posisi yang diberikan kepada JST pada saat belajar, maka jawaban estimasi JST pada saat penggunaannya akan semakin baik. Untuk pengenalan posisi yang diskrit penggunaan JST akan lebih baik dengan cara mengajari JST untuk memberi jawaban yang sama untuk sekelompok pola posisi dalam daerah yang berdekatan. Pada saat penggunaannya JST akan memberi jawaban yang sama untuk setiap posisi yang masih termasuk dalam kelompok yang sama.

Pengujian kemampuan Jaringan Syaraf Tiruan mengatasi derau dilakukan dengan memberikan sinar lampu di depan sensor infra merah, dimana sinar lampu tersebut juga mengandung sinar infra merah. Untuk beberapa posisi yang sudah diajarkan kepada JST direkam pada kondisi berderau, kemudian dipropagasi maju untuk mendapatkan tanggapan JST.

Perbandingan respon JST pada kondisi tidak berderau dan kondisi berderau

dapat dilihat pada Tabel 5.3 dan Tabel 5.4. Dan ternyata JST masih dapat mengenali pola masukan yang berderau, walaupun jawaban JST tidak benar-benar sama dengan jawaban dalam kondisi tanpa derau. Salah satu cara agar JST lebih mampu mengatasi derau adalah mengajarinya juga dengan pola-pola yang sudah bercampur derau.

**Tabel 5.1 Pembandingan nilai fungsi estimasi JST dengan sesungguhnya setelah galat setiap sel output pada saat belajar sekitar 1E-3.**

| No. | Posisi Yang Sebenarnya (X,Y) | Hasil Estimasi Posisi JST (X,Y) |
|---|---|---|
| 1. | (0.000 , 2.000) | (-0.0170 , -2.6917) |
| 2. | (0.000 , 3.000) | (0.0833 , 3.0257) |
| 3. | (1.000 , 3.000) | (0.0314 ,3.0987) |
| 4. | (-1.000 , 3.000) | (-0.0355 , 3.4199) |
| 5. | (0.000 , 4.000) | (0.1516 , 3.5094) |
| 6. | (5.000 , 3.000) | (4.8870 , 3.8015) |
| 7. | (4.000 , 3.000) | (4.8000 , 4.1187) |
| 8. | (-5.000 , 3.000) | (-5.0577 , 3.3184) |
| 9. | (-4.000 , 3.000) | (-4.7398 , 3.7066) |
| 10. | (0.000 , 9.000) | (-0.0750 , 10.2303) |
| 11. | (-5.000 , 9.000) | (-4.9410 , 9.5953) |
| 12. | (1.000 , 15.000) | (1.0128 , 15.6606) |
| 13. | (0.000 , 15.000) | (0.0098 , 15.0025) |
| 14. | (0.000 , 17.000) | (0.0452 , 16.2373) |
| 15. | (0.000 , 23.000) | (0.1611 , 22.3453) |
| 16. | (10.000 , 30.000) | (-6.1030 , -1.0413) |
| 17. | (-12.000 , 3.000) | (-4.5882 , 2.2036) |

**Tabel 5.1 Pembandingan nilai fungsi estimasi JST dengan sesungguhnya setelah galat setiap sel output pada saat belajar sekitar 1E-5.**

| No. | Posisi Yang Sebenarnya (X,Y) | Hasil Estimasi Posisi JST (X,Y) |
|---|---|---|
| 1. | (0.000 , 2.000) | (0.3788 , -10.6739) |
| 2. | (0.000 , 3.000) | (0.0003 , 3.0015) |
| 3. | (1.000 , 3.000) | (-0.1911 , 9.2590) |
| 4. | (-1.000 , 3.000) | (-0.1835 , 3.9421) |
| 5. | (0.000 , 4.000) | (0.4383 , 4.2503) |
| 6. | (5.000 , 3.000) | (4.8870 , 3.8015) |
| 7. | (4.000 , 3.000) | (4.8000 , 4.1187) |
| 8. | (-5.000 , 3.000) | (-5.0052 , 3.0031) |
| 9. | (-4.000 , 3.000) | (-4.7398 , 13.7066) |
| 10. | (0.000 , 9.000) | (0.0006 , 9.0004) |
| 11. | (-5.000 , 9.000) | (-4.9995 , 9.0003) |
| 12. | (1.000 , 15.000) | (2.0348 , 10.0076) |
| 13. | (0.000 , 15.000) | (0.0003 , 15.0006) |
| 14. | (0.000 , 17.000) | (2.1435 , 15.1024) |
| 15. | (0.000 , 21.000) | (0.0005 , 21.0023) |
| 16. | (10.000 , 30.000) | (23.0342 , 15.8897) |
| 17. | (-12.000 , 3.000) | (15.0730 , 20.2341) |

**Tabel 5.3 JST diberikan pola masukan yang tidak ditambahi noise.**

| No. | Posisi Yang Sebenarnya (X,Y) | Posisi Hasil Estimasi JST (X,Y) |
|---|---|---|
| 1. | (0.000 , 3.000) | (0.0029 , 3.0006) |
| 2. | (-5.000 , 3.000) | (-4.9852 ,3.0006) |
| 3. | (5.000 , 3.000) | (5.0243 , 3.0143) |
| 4. | (5.000 , 9.000) | (4.8695 , 9.0346) |
| 5. | (0.000 , 9.000) | (-0.0265 , 8.9431) |
| 6. | (-5.000 , 9.000) | (-4.9733 , 9.0279) |
| 7. | (5.000 , 15.000) | (-5.2731, 14.7968) |
| 8. | (0.000 , 15.000) | (0.1312 , 15.2856) |
| 9. | (-5.000 , 15.000) | (5.3264 , 14.9184) |
| 10. | (5.000 , 21.000) | (4.8831 , 21.0084) |
| 11. | (0.000 , 21.000) | (-0.2469, 20.9838) |
| 12. | (-5.000 , 21.000) | (-4.5880, 21.0881) |

Tabel 5.4 **JST diberikan pola masukan yang ditambahi noise berupa cahaya.**

| No. | Posisi Yang Sebenarnya (X,Y) | Posisi Hasil Estimasi JST (X,Y) |
|---|---|---|
| 1. | (0.000 , 3.000) | (0.3207 , 3.1896) |
| 2. | (-5.000 , 3.000) | (-4.4350 ,3.1505) |
| 3. | (5.000 , 3.000) | (4.9477 , 3.2639) |
| 4. | (5.000 , 9.000) | (6.4713 , 8.7221) |
| 5. | (0.000 , 9.000) | (0.5157 , 8.7478) |
| 6. | (-5.000 , 9.000) | (-5.4808 , 7.9851) |
| 7. | (5.000 , 15.000) | (-3.6093 , 16.0983) |
| 8. | (0.000 , 15.000) | (0.3052 , 15.5078) |
| 9. | (-5.000 , 15.000) | (6.0213 , 15.4192) |
| 10. | (5.000 , 21.000) | (2.8640 , 19.6153) |
| 11. | (0.000 , 21.000) | (-0.2295 , 19.1290) |
| 12. | (-5.000 , 21.000) | (-2.4304 , 20.4004) |

# BAB VI

# KESIMPULAN DAN TINDAK LANJUT

## 6.1 Kesimpulan

Jaringan Syaraf Tiruan dapat dimanfaatkan pada pemroses sinyal sensor sistem docking. Pemanfaatan tersebut dimungkinkan karena sifat mampu belajar yang dimiliki oleh Jaringan Syaraf Tiruan. Dengan menggunakan Jaringan Syaraf Tiruan sebagai pemroses sinyal sensor, [illegible] matematis yang sulit pada perhitungan hubungan masukan keluaran dari fungsi yang diinginkan menjadi lebih sederhana.

Dari uji coba yang [illegible] beberapa hal

1. Jaringan Syaraf [illegible] sistem yang mampu belajar. Kemampuan belajar Jaringan [illegible] kemudahan dalam membentuk fungsi pemetaan pola-pola masukan sensor menjadi informasi keluaran yang [illegible]
2. Untuk membuat [illegible] pemetaan pola-pola [illegible] Jaringan Feed Forward membagi daerah [illegible] keputusan.
3. Kemampuan Jaringan [illegible] mengenal pola-pola masukan (dalam hal ini posisi sensor terhadap target) sangat tergantung pada jumlah posisi yang dimasukkan dalam paket belajar. Semakin banyak pola yang

# BAB VI

# KESIMPULAN DAN TINDAK LANJUT

## 6.1 Kesimpulan

Jaringan Syaraf Tiruan dapat dimanfaatkan pada pemroses sinyal sensor sistem docking. Pemanfaatan tersebut dimungkinkan karena sifat mampu belajar yang dimiliki oleh Jaringan Syaraf Tiruan. Dengan menggunakan Jaringan Syaraf Tiruan sebagai pemroses sinyal sensor, masalah matematis yang sulit pada perhitungan hubungan masukan keluaran dari fungsi yang diinginkan menjadi lebih sederhana.

Dari uji coba yang dilakukan dapat disimpulkan beberapa hal

1. Jaringan Syaraf Tiruan merupakan sistem yang mampu belajar. Kemampuan belajar Jaringan Syaraf Tiruan akan memberikan kemudahan dalam membentuk fungsi pemetaan pola-pola masukan sensor menjadi informasi keluaran yang diinginkan.
2. Untuk membuat suatu pemetaan pola-pola masukan Jaringan **Feed Forward** membagi daerah masukan menjadi beberapa daerah keputusan.
3. Kemampuan Jaringan Syaraf Tiruan untuk mengenal pola-pola masukan (dalam hal ini posisi sensor terhadap target) sangat tergantung pada jumlah posisi yang dimasukkan dalam paket belajar. Semakin banyak pola yang

diajarkan semakin teliti juga hasil estimasi JST terhadap pola masukan pada saat penggunaannya.

4. Kemampuan Jaringan Syaraf Tiruan akan bertambah jika jumlah sel-sel pada lapisan dalam ditambah.

Untuk mengaplikasikan JST pada sistem docking ini diperlukan penentuan struktur jaringan yang meliputi jumlah sel dan jumlah lapis, begitu juga pembuatan paket belajar dan pelatihan jaringan.

## 6.2 Tindak Lanjut

Dari penelitian yang telah dilakukan menunjukkan bahwa, pengolah sinyal sensor yang memetakan pola masukan menjadi keluaran dapat dibentuk dengan Jaringan Syaraf Tiruan. Namun demikian kemampuannya masih terbatas. Untuk meningkatkan kemampuan ini, salah satu cara yang dapat dilakukan adalah dengan diberikan lebih banyak lagi pola belajar. Jaringan Syaraf Tiruan tersebut harus diberikan juga pola belajar untuk kondisi yang berderau, sehingga setiap masukan dari sensor akan diberi jawaban yang benar tidak perduli apakah berderau atau tidak.

Untuk merealisasikan Sistem Docking secara keseluruhan, masih diperlukan pengontrol utama, penggerak cerdas dan bagian-bagian lain yang mendukungnya. Karena yang dibentuk JST dalam penelitian ini hanya ditujukan untuk memetakan

setiap pola masukan sensor menjadi variabel yang dibutuhkan untuk pengambilan keputusan oleh pengontrol utama.

Penelitian lebih lanjut terhadap perangkat lainnya maupun bagian pengolah sinyal sensor ini, diharapkan dapat digunakan untuk merealisasikan sistem docking.

# DAFTAR PUSTAKA


[1] Ad Dairi,S.H, *Karakteristik Jaringan Syaraf Tiruan dengan Arsitektur Feedforward*, Jurusan Teknik Elektro, ITB, 1991.

[2] de Villiers, Jacques and Barnard, Etienne, **Backpropagation Neural Nets with One and Two Hidden Layers**, *IEEE Transaction on Neural Networks*, pp. 136-141, Vol.4 No.1, Jan 1992.

[3] Duda, Richard O., *Pattern Classification and Scene Analysis*, John Wiley and Sons, Inc., Toronto, 1973.

[4] Freeman, James A. and Skapura, David M., *Neural Networks Algorithms, Applications, and Programming Techniques*, Addison-Wesley Publishing Company, Inc., [illegible], 1991.

[5] Pao, Y.H., *Adaptive [illegible] Neural Networks*, Addison-Wesley, Massachusetts, 1989.

[6] Wessels, Lodewyk [illegible] and Barnard, [illegible] **Avoiding False Local Minima by Proper [illegible]ization of Connections**, *IEEE Transactions on Neural Networks*, pp. [illegible], Vol.3 No.6, Nov 1992.

[7] Yu, Xiao-Hu, **Can Backpropagation Error Surface Not Have Local Minima**, *IEEE Transaction on Neural Networks*, pp. 1019-1021, Vol.3 No.6, Nov 1992.

# DAFTAR PUSTAKA


[1] Ad Dairi,S.H, ***Karakteristik Jaringan Syaraf Tiruan dengan Arsitektur Feedforward***, Jurusan Teknik Elektro, ITB, 1991.

[2] de Villiers, Jacques and Barnard, Etienne, **Backpropagation Neural Nets with One and Two Hidden Layers**, ***IEEE Transaction on Neural Networks***, pp. 136-141, Vol.4 No.1, Jan 1992.

[3] Duda, Richard O., ***Pattern Classification and Scene Analysis***, John Wiley and Sons, Inc., Toronto, 1973.

[4] Freeman, James A. and Skapura, David M., ***Neural Networks Algorithms, Applications, and Programming Techniques***, Addison-Wesley Publishing Company, Inc., California, 1991.

[5] Pao, Y.H., ***Adaptive Pattern Recognition and Neural Networks***, Addison-Wesley, Massachusettes, 1989.

[6] Wessels, Lodewyk F.A. and Barnard, Etienne, **Avoiding False Local Minima by Proper Initialization of Connections**, ***IEEE Transactions on Neural Networks***, pp. 899-905, Vol.3 No.6, Nov 1992.

[7] Yu, Xiao-Hu, **Can Backpropagation Error Surface Not Have Local Minima**, ***IEEE Transaction on Neural Networks***, pp. 1019-1021, Vol.3 No.6, Nov 1992.